

When is Clustering Perturbation Robust?

Margareta Ackerman and Jarrod Moore

Abstract

Clustering is a fundamental data mining tool that aims to divide data into groups of similar items. Generally, intuition about clustering reflects the ideal case – exact data sets endowed with flawless dissimilarity between individual instances.

In practice however, these cases are in the minority, and clustering applications are typically characterized by noisy data sets with approximate pairwise dissimilarities. As such, the efficacy of clustering methods in practical applications necessitates robustness to perturbations.

In this paper, we perform a formal analysis of perturbation robustness, revealing that the extent to which algorithms can exhibit this desirable characteristic is inherently limited, and identifying the types of structures that allow popular clustering paradigms to discover meaningful clusters in spite of faulty data.

1 Introduction

Clustering is a popular data mining tool, due in no small part to its general and intuitive goal of dividing data into groups of similar items. Yet in spite of this seemingly simple task, successful application of clustering techniques in practice is oftentimes challenging. In particular, there are inherent difficulties in the data collection process and design of pairwise dissimilarity measures, both of which significantly impact the behavior of clustering algorithms.

Intuition about clustering often reflects the ideal case – flawless data sets with well-suited dissimilarity between individual instances. In practice however, these cases are rare. Errors are introduced into a data set for a wide variety of reasons; from precision of instruments (a student’s ruler to the Large Hadron Collider alike have a set precision), to human error when data is user-reported (common in the social sciences). Additionally, the dissimilarity between pairwise instances is often based on heuristic measures, particularly when non-numeric attributes are present. Furthermore, the dynamic nature of prominent clustering applications (such as personalization for recommendation systems) implies that by the time the data has been clustered, it has already changed.

The ubiquity of flawed input poses a serious challenge. If clustering is to operate strictly under the assumption of ideal data, its applicability would be reduced to fairly rare applications where such data can be attained. As such, it would be desirable for clustering algorithms to provide some qualitative guarantees about their output when partitioning noisy data. This leads us to explore whether there are any algorithms for which such guarantees can be provided.

Although data can be faulty in a variety of ways, our focus here is on inaccuracies of pairwise distances. At a minimum, small perturbation to data should not radically affect the output of an algorithm. It would be natural to expect that some clustering techniques are more robust than others, allowing users to rely on perturbation robust techniques when pairwise distances are inexact.

However, our investigation reveals that no reasonable clustering algorithm exhibits this desirable characteristic. In fact, both additive and multiplicative perturbation robustness are unrealistic requirements. We show that no clustering algorithm can satisfy robustness to perturbation without violating even more fundamental requirements. Not only do existing methods lack this desirable characteristic, but our findings also preclude the possibility of designing novel perturbation robust clustering methods.

Perhaps it is already surprising that no reasonable clustering algorithm can be perfectly perturbation robust, but our results go further. Instead of requiring that the clustering remain unchanged following a

perturbation, we allow up to *two-thirds* of all pairwise distances to change (from in-cluster to between-cluster, or vice-versa). It turns out that this substantial relaxation doesn't overcome our impossibility theorem.

Luckily, further exploration paints a more optimistic picture. A careful examination of this issue requires a look back to the underlying goal of clustering, which is to discover clustering structure in data *when such structure is present*. Our investigation suggests that sensitivity to small perturbations is inevitable only on unclusterable instances, for which clustering is inherently ill-suited. As such, it can be argued that whether an algorithm exhibits robustness on such data is inconsequential.

On the other hand, we show that when data is endowed with inherent structure, existing methods can often successfully reveal that structure even on faulty (perturbed) data. We investigate the type of cluster structures required for the success of popular clustering techniques, showing that the robustness of k -means and related methods is directly proportional to the degree of inherent cluster structure. Similarly, we show that popular linkage-based techniques are robust when clusters are well-separated. Furthermore, different cluster structures are necessary for different algorithms to exhibit robustness to perturbations.

1.1 Previous work

This work follows a line of research on theoretical foundations of clustering. Efforts in the field began as early as the 1970s with the pioneering work of Wright [1973] on axioms of clustering, as well as analysis of clustering properties by Fisher [1971] and Van Ness [1971] and Jardine and Sibson [1971], among others. This field saw a renewed surge of activity following Kleinberg's [2003] famous impossibility theorem, when he showed that no clustering function can simultaneously satisfy three simple properties. Also related to our work is a framework for selecting clustering methods based on differences in their input-output behavior Ackerman et al. [2010b, 2012], Jardine and Sibson [1971], Zadeh and Ben-David [2009], Ackerman et al. [2010a, 2013] as well as research on clusterability, which aims to quantify the degree of inherent cluster structure in data Ben-David [2015], Ackerman and Ben-David [2009], Balcan et al. [2008a,b], Ostrovsky et al. [2006].

Previous work on perturbation robustness studies it from a computational perspective by identifying new efficient algorithms for robust instances Bilu and Linial [2010], Ackerman and Ben-David [2009], Awasthi et al. [2012]. Ben-David and Reyzin [2014] recently studied corresponding NP-hardness lower bounds.

In this paper, we take a fresh look at perturbation robustness. We begin our investigation by asking when perturbation robustness is possible to attain. After proving that robustness to perturbations cannot be achieved as a data-independent property of an algorithm, we seek to understand when popular clustering paradigms satisfy this requirement. Our analysis of established methods is an essential complement to efforts in algorithmic development, as the need for understanding established methods is amplified by the fact that most clustering users rely on a small number of well-known techniques. Our results demonstrate the type of cluster structures required for robustness of popular clustering paradigms.

2 Definitions and notation

Clustering is a wide and heterogeneous domain. For most of this paper, we focus on a basic sub-domain where the input to a clustering function is a finite set of points endowed with a between-points dissimilarity function, and the number of clusters (k), and the output is a partition of that domain.

A *dissimilarity function* is a symmetric function $d : X \times X \rightarrow R^+$, such that $d(x, x) = 0$ for all $x \in X$. The data sets that we consider are pairs (X, d) , where X is some finite domain set and d is a dissimilarity function over X .

A k -*clustering* $C = \{C_1, C_2, \dots, C_k\}$ of a data set X is a partition of X into k disjoint subsets (or, clusters) of X (so, $\bigcup_i C_i = X$). A *clustering* of X is a k -clustering of X for some $1 \leq k \leq |X|$.

For a clustering C , let $|C|$ denote the number of clusters in C and $|C_i|$ denote the number of points in a cluster C_i . For a domain X , $|X|$ denotes the number of points in X , which we denote by n when the domain

is clear from context. We write $x \sim_C y$ if x and y are both in some cluster C_j ; and $x \not\sim_C y$ otherwise. This is an equivalence relation.

The *Hamming distance* between clusterings C and C' of the same domain set X is defined by

$$\Delta(C, C') = \frac{|\{\{x, y\} \subset X \mid (x \sim_C y) \oplus (x \sim_{C'} y)\}|}{\binom{|X|}{2}},$$

where \oplus denotes the logical XOR operation.

That is, the difference is the number of edges that disagree, being in-cluster in one of the clusterings and between-cluster in the other. The maximum distance between clusterings is when the Hamming distance is 1.

Lastly, we formally define clustering functions.

Definition 1 (Clustering function). *A clustering function is a function \mathcal{F} that takes as input a pair (X, d) and a parameter $1 \leq k \leq |X|$, and outputs a k -clustering of the domain X .*

3 Perturbation robustness as a property of an algorithm

Whenever a user is faced with the task of clustering faulty data, it would be natural to select an algorithm that is robust to perturbations of pairwise dissimilarities. As such, we begin our study of perturbation robustness by casting it as a property of an algorithm. If we could classify algorithms based on whether or not (or to what degree) they are perturbation robust, then clustering users could incorporate this information when making decisions regarding which algorithms to apply on their data.

First, we define what it means to perturb a dissimilarity function.

Definition 2 (α -multiplicative perturbation of a dissimilarity function). *Given a pair of dissimilarity functions d and d' over a domain X , d' is an α -multiplicative-perturbation of d , for $\alpha > 1$, if for all $x, y \in X$, $\alpha^{-1}d(x, y) \leq d'(x, y) \leq \alpha d(x, y)$.*

Additive perturbation of a dissimilarity function is defined analogously.

Definition 3 (ϵ -additive perturbation of a dissimilarity function). *Given a pair of dissimilarity functions d and d' over a domain X , d' is an ϵ -additive perturbation of d , for $\epsilon > 0$, if for all $x, y \in X$, $d(x, y) - \epsilon \leq d'(x, y) \leq d(x, y) + \epsilon$.*

It is important to note that all of our results hold for both multiplicative and additive perturbation robustness. Perturbation robust algorithms should be invariant to data perturbations; that is, if data is perturbed, then the output of the algorithm shouldn't change. This view of perturbation robustness is not only intuitive, but is also based on previous formulations Reyzin [2012], Bilu and Linial [2010], Awasthi et al. [2012] (This can be formalized as a property of clustering functions by setting $\delta = 0$ in Definition 4 below).

However, requiring that the partitioning be identical before and after perturbation is provably too strict a requirement for clustering algorithms, as it can only hold for functions that effectively ignore all pairwise distances. That is, this notion of perturbation robustness only holds for clustering functions that, given any domain set X and integer $1 \leq k \leq |X|$, produce the same partitioning regardless of the setting of d (See Section 1 of the Appendix for details).

As such, we introduce a relaxation that allows some error in the output of the algorithm on perturbed data. From a practical point of view, it is likely that a user who has only a perturbation of the true data set is likely to be satisfied with an approximately correct solution.

Definition 4. *A clustering function \mathcal{F} is (α, δ) -multiplicative perturbation robust if, given any data set (X, d) and $1 \leq k \leq |X|$, whenever d' is an α -multiplicative perturbation of d ,*

$$\Delta(\mathcal{F}(X, d, k), \mathcal{F}(X, d', k)) < \delta.$$

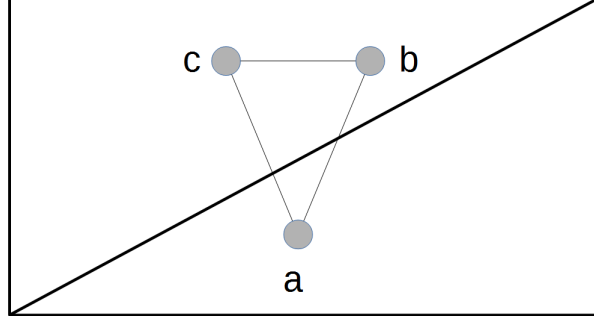


Figure 1: An illustration of the three-body rule. Based on the main objective of clustering, which is to group similar items together, this rule requires that whenever an algorithm is given exactly three points and the number of clusters is two, then it groups the two closest elements.

Additive perturbation robustness is defined analogously, by replacing the α -multiplicative perturbation of the dissimilarity function with an ϵ -additive perturbation.

Despite this substantial relaxation, perturbation robustness for δ as high as $2/3$ is inherently incompatible with even more elementary requirements, as shown below.

3.1 Impossibility theorem for clustering functions

We now proceed to show that perturbation robustness is too strong a requirement for clustering algorithms, and as such neither existing nor novel techniques can have this desirable characteristic.

Particularly notable is that the impossibility results persists when δ is as high as $2/3$, meaning that a perturbation is allowed to change up to two-thirds of all pairwise distances from in-cluster to between-cluster, or vice-versa. As such, we show that no reasonable clustering algorithm can preserve more than a third of its pairwise distances after a perturbation.

The following impossibility result derives from the pioneering work of Wright Wright [1973] on axioms of clustering. Wright originally proposed his axioms in Euclidean space, here we generalize them for arbitrary pairwise dissimilarities.

The first axiom we discuss follows from Wright’s 11th axiom, and captures the very essence of clustering: to group similar items. This property considers an elementary scenario, requiring that given exactly three points, an algorithm asked for two clusters should group the two closest elements. See Figure 1 for an illustration. A special case of this rule occurs when the three elements lie on the real line, in which case the furthest endpoint should be placed in its own cluster.

Definition 5 (Three-body rule). *Given a data set $X = \{a, b, c\}$, if $d(a, b) > d(b, c)$ and $d(a, c) > d(b, c)$, then $\mathcal{F}(X, d, 2) = \{\{a\}, \{b, c\}\}$.*

Wright’s 6th axiom, and the final one we consider here, requires that replicating all data points by the same number should not change the clustering output. Outside of Euclidean space, we *replicate* a point x by adding a new element x' and setting $d(x', y) = d(x, y)$ for all $y \in X$.

Definition 6 (Replication invariance). *Given any positive integer r , if all points are replicated r times, then the partitioning of the original data is unchanged and all replicas lie in the same cluster as their original element.*

Not only are these two axioms natural, as violating them leads to counterintuitive behavior, but they also hold for common techniques. It is easy to show that they are satisfied by common clustering paradigms, including cost-based methods such as k -means, k -median, and k -medoids, as well as linkage-based techniques, such as single-linkage, average-linkage and complete-linkage.

We now prove that no clustering function that satisfies the three-body rule and replication invariance can be perturbation robust. Furthermore, our result holds for all values of $\delta \leq 2/3$. Note that the following result applies to arbitrarily large data sets, for both multiplicative and additive perturbations.

Theorem 1. *For any $\delta \leq 2/3$, $\alpha > 1$, and $\epsilon > 0$, there is no clustering function that satisfies*

1. *(α, δ) -multiplicative perturbation robustness, replication invariance, and the three-body rule, and*
2. *(ϵ, δ) -additive perturbation robustness, replication invariance, and the three-body rule.*

Further, the result holds for arbitrarily large data.

Proof. We proceed by contradiction, assuming that there exists a clustering function \mathcal{F} that is replication invariant, adheres to the three-body rule, and is (α, δ) -multiplicative perturbation robust for some $\delta \leq 2/3$.

Consider a data set $X = \{a, b, c\}$ with a distance function d such that $d(b, c) < d(a, b) < d(a, c)$ and $d(a, b) = \alpha d(b, c)$. By the three-body rule, $\mathcal{F}(X, d, 2) = \{\{b, c\}, \{a\}\}$. We now replicate each point an arbitrary number of times, r , creating three sets A, B, C such that all points that are replicas of the point a and a itself belong to A , all points that are replications of the point b and b itself belong to B , and similarly for C . By replication invariance, $\mathcal{F}(A \cup B \cup C, d, 2) = \{B \cup C, A\}$.

Next, we apply an α -multiplicative perturbation, creating distance function d' such that $d'(a, b) < d'(b, c) < d'(a, c)$ and $d'(c, b) = \alpha d'(b, a)$. By the three-body rule, $\mathcal{F}(A \cup B \cup C, d', 2) = \{B \cup A, C\}$, and yet $(\alpha, 2/3)$ -multiplicative perturbation robustness requires that the Hamming distance between $\mathcal{F}(A \cup B \cup C, d, 2)$ and $\mathcal{F}(A \cup B \cup C, d', 2)$ must be less than $2/3$. But as the Hamming distance between $\{B \cup C, A\}$ and $\{B \cup A, C\}$ is exactly $2/3$, we reach a contradiction.

For additive perturbation, set d so that $d(b, c) < d(a, b) < d(a, c)$ and $d(a, b) = d(b, c) + 0.5\epsilon$. By the three-body rule, $\mathcal{F}(X, d, 2) = \{\{b, c\}, \{a\}\}$. As for the multiplicative case, we replicate each point r times, creating three sets A, B, C . By replication invariance, $\mathcal{F}(A \cup B \cup C, d, 2) = \{B \cup C, A\}$. We apply an ϵ -additive perturbation to make distance function d' such that $d'(a, b) < d'(b, c) < d'(a, c)$ and $d'(c, b) = d'(b, a) + 0.5\epsilon$. By the three-body rule, $\mathcal{F}(A \cup B \cup C, d', 2) = \{B \cup A, C\}$, and yet $(\alpha, 2/3)$ -additive perturbation robustness requires that the Hamming distance between $\mathcal{F}(A \cup B \cup C, d, 2)$ and $\mathcal{F}(A \cup B \cup C, d', 2)$ must be less than $2/3$, reaching a contradiction. \square

Note that the above result holds if the data is in Euclidean space. This allows us to view perturbations as small movements in space, required to satisfy certain constraints such as the triangle inequality as well as adhering to the dissimilarity constraints required by Definitions 2 and 3. See supplementary material for details.

4 Perturbation robustness as a property of data

The above section demonstrates an inherent limitation of perturbation robustness as a property of clustering algorithms, showing that no reasonable clustering algorithm can exhibit this desirable characteristic. However, it turns out that perturbation robustness is possible to achieve when we restrict our attention to data endowed with inherent structure.

As such, perturbation robustness becomes a property of both an algorithm and a specific data set. We introduce a definition of perturbation robustness that directly addresses the underlying data.

Definition 7 *$((\alpha, \delta)$ -multiplicative perturbation robustness of data). A data set (X, d) satisfies (α, δ) -multiplicative perturbation robustness with respect to clustering function \mathcal{F} and $1 \leq k \leq |X|$, if for any d' that is an α -multiplicative perturbation of d ,*

$$\Delta(\mathcal{F}(X, d, k), \mathcal{F}(X, d', k)) < \delta.$$

Additive perturbation robustness of data is defined analogously.

This perspective at perturbation robustness raises a natural question: On what types of data are algorithms perturbation robust? Next, we explore the type of structures that allow popular cost-based paradigms and linkage-based methods to uncover meaningful clusters even when data is faulty.

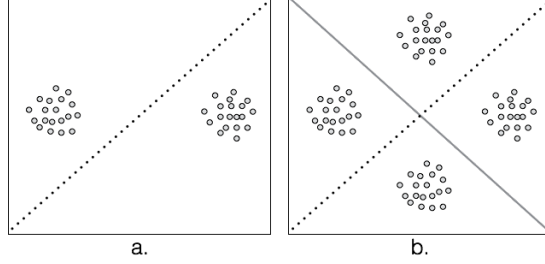


Figure 2: An illustration of the uniqueness of optimum notion of clusterability for two clusters. Consider k -means, k -medoids, or min-sum. The highly-clusterable data depicted in (a) has a unique optimal solution, with no structurally different clusterings of near-optimal cost. In contrast, (b) displays data with two radically different clusterings of near-optimal cost, making this data poorly-clusterable for $k = 2$.

4.1 Perturbation robustness of k -means, k -medoids, and min-sum

We begin our study of data-dependent perturbation robustness by considering cluster structures required for perturbation robustness of some of the most popular clustering functions: k -means, k -medoids and min-sum.

Recall that k -means Steinley [2006] finds the clustering $C = \{C_1, \dots, C_k\}$ that minimizes $\sum_{i=1}^k \sum_{x \in C_i} d(x, c_i)^2$, where c_i is the center of mass of cluster C_i . An equivalent formulation that does not rely on centers of mass appears in Ostrovsky et al. [2006]. A closely related clustering function is k -medoids, where centers are required to be part of the data. Formally, the k -medoids cost of C is $\sum_{i=1}^k \sum_{x \in C_i} d(x, c_i)$, where $c_i \in C_i$ is chosen to minimize the objective. Lastly, the min-sum Sahni and Gonzalez [1976] clustering function is the sum of all in-cluster distances, $\sum_{i=1}^k \sum_{x, y \in C_i} d(x, y)$.

Many different notions of clusterability have been proposed in prior work Ackerman and Ben-David [2009], Ben-David [2015]. Although they all aim to quantify the same tendency, it has been proven that notions of clusterability are often pairwise inconsistent Ackerman and Ben-David [2009]. As such, care must be taken when selecting amongst them.

In order to analyze k -means and related functions, we turn our attention to an intuitive cost-based notion, which requires that clusterings of near-optimal cost be structurally similar to the optimal solution. That is, this notion characterizes clusterable data as that which has a unique optimal solution in a strong sense, by excluding the possibility of having radically different clusterings of similar cost. See Figure 2 for an illustration.

This property, called “uniqueness of optimum”¹ and closely related variations were investigated by Balcan et al. [2008a], Ostrovsky et al. [2006], Agarwal et al. [2013] and Ackerman et al. [2013], among others. See Balcan et al. [2008a] for a detailed exposition.

Definition 8 (Uniqueness of optimum). *Given a clustering function \mathcal{F} , a data set (X, d) is (δ, c, c_0, k) -uniquely optimal if for every k -clustering C of X where $\text{cost}(C) \leq c \cdot \text{cost}(\mathcal{F}(X, d, k)) + c_0$,*

$$\Delta(\mathcal{F}(X, d, k), C) < \delta.$$

We show that whenever data satisfies the uniqueness of optimum notion of clusterability, k -means, k -medoids, and min-sum are perturbation robust. Furthermore, the degree of robustness depends on the extent to which the data is clusterable.

For the following proofs we will use $\text{cost}_d(C)$ to denote the cost of clustering C with the distance function d . We now show the relationships between uniqueness of optimum and perturbation robustness for k -means.

¹This notion of clusterability appeared under several different names. The term “uniqueness of optimum” was coined by Ben-David Ben-David [2015].

Theorem 2. Consider the k -means clustering function and a data set (X, d) . If (X, d) is (δ, c, c_0, k) -uniquely optimal, then it is also (ϵ, δ, k) -additive perturbation robust for all $\epsilon < \min(\frac{c-1}{2}, \frac{-M + \sqrt{M^2 + 4MC_0}}{2M})$, where $M = \binom{n}{2}$.

Proof. Consider a data set (X, d) , and let d' be any ϵ -additive perturbation of d .

First, we argue that $\text{cost}_{d'}(\mathcal{F}(X, d, k))$ is close to $\text{cost}_{d'}(\mathcal{F}(X, d', k))$. Let $C = \mathcal{F}(X, d, k)$. First, note that $\text{cost}_{d'}(\mathcal{F}(X, d', k)) \leq \text{cost}_{d'}(C)$. This is because \mathcal{F} finds the optimal solution on (X, d') , and so the clustering it selects can only have lower or equal to cost than the cost of C when evaluated with d' .

So, we calculate the k -means cost of C on (X, d') . The k -means objective function is equivalent to $\sum_{i=1}^k \frac{1}{|C_i|} \sum_{x, y \in C_i} d(x, y)^2$ Ostrovsky et al. [2006]. After an additive perturbation, any pairwise distance, $d(x, y)$, is bounded by $d(x, y) + \epsilon$. In addition, the contribution of any in-cluster pairwise distance to the total cost of the clustering is proportional to the magnitude of the distance. It therefore follows

$$\text{cost}_{d'}(\mathcal{F}(X, d', k)) \leq \text{cost}_{d'}(C) \quad (1a)$$

$$\leq \sum_{i=1}^k \frac{1}{|C_i|} \sum_{\{x, y\} \subseteq C_i} (d(x, y) + \epsilon)^2 \quad (1b)$$

$$\leq \sum_{i=1}^k \sum_{\{x, y\} \subseteq C_i} \frac{1}{|C_i|} (d(x, y) + \epsilon)^2 \quad (1c)$$

$$\leq \sum_{i=1}^k \sum_{\{x, y\} \subseteq C_i} \frac{1}{|C_i|} [d(x, y)^2 + 2d(x, y)\epsilon + \epsilon^2] \quad (1d)$$

By distributing the summation in the inequality 1d we come to:

$$\text{cost}_{d'}(\mathcal{F}(X, d', k)) \leq \sum_{i=1}^k \sum_{\{x, y\} \subseteq C_i} \frac{1}{|C_i|} d(x, y)^2 \quad (2a)$$

$$+ \sum_{i=1}^k \sum_{\{x, y\} \subseteq C_i} \frac{1}{|C_i|} 2d(x, y)\epsilon \quad (2b)$$

$$+ \sum_{i=1}^k \sum_{\{x, y\} \subseteq C_i} \frac{1}{|C_i|} \epsilon^2 \quad (2c)$$

The first term, 2a, is equivalent to $\text{cost}_d(\mathcal{F}(X, d, k))$. We deal with the second term, 2b, by defining two sets S_1 and S_2 . To define S_1 , we first define S_{1i} . $S_{1i} = \{\{x, y\} \subseteq C_i | d(x, y) > 1\}$. Then $S_1 = \{S_{1i} | 1 \leq i \leq k\}$. Similarly $S_{2i} = \{\{x, y\} \subseteq C_i | d(x, y) \leq 1\}$, and $S_2 = \{S_{2i} | 1 \leq i \leq k\}$.

$$\sum_{i=1}^k \sum_{\{x, y\} \subseteq C_i} \frac{1}{|C_i|} 2d(x, y)\epsilon \leq \sum_{i=1}^k \sum_{\{x, y\} \in S_{1i}} \frac{1}{|C_i|} 2d(x, y)\epsilon \quad (3a)$$

$$+ \leq \sum_{i=1}^k \sum_{\{x, y\} \in S_{2i}} \frac{1}{|C_i|} 2d(x, y)\epsilon \quad (3b)$$

Because for all $\{x, y\} \in S_{1i}$ for all $1 \leq i \leq k$, $d(x, y) > 1$, we can square the $d(x, y)$ value in term 3a while only increasing the total value. Likewise, we can replace the $d(x, y)$ value in term 3b with 1 while only increasing the total value.

$$\sum_{i=1}^k \sum_{\{x,y\} \subseteq C_i} \frac{1}{|C_i|} 2d(x,y)\epsilon \leq \sum_{i=1}^k \sum_{\{x,y\} \in S_{1i}} \frac{1}{|C_i|} 2d(x,y)^2\epsilon \quad (4a)$$

$$+ \sum_{i=1}^k \sum_{\{x,y\} \in S_{2i}} \frac{1}{|C_i|} 2\epsilon \quad (4b)$$

Since S_{1i} and S_{2i} both consist of point pairs in C_i and we are looking for an upper bound:

$$\sum_{i=1}^k \sum_{\{x,y\} \subseteq C_i} \frac{1}{|C_i|} 2d(x,y)\epsilon \leq \sum_{i=1}^k \sum_{\{x,y\} \subseteq C_i} \frac{1}{|C_i|} 2d(x,y)^2\epsilon \quad (5a)$$

$$+ \sum_{i=1}^k \sum_{\{x,y\} \subseteq C_i} \frac{1}{|C_i|} 2\epsilon \quad (5b)$$

Note that $\sum_{i=1}^k \sum_{\{x,y\} \subseteq C_i} \frac{1}{|C_i|} 2d(x,y)^2\epsilon$ is equivalent to $2\epsilon \text{cost}_d(\mathcal{F}(X, d))$. We can now return to the original inequality.

$$\text{cost}_{d'}(\mathcal{F}(X, d', k)) \leq \text{cost}_d(\mathcal{F}(X, d, k)) \quad (6a)$$

$$+ 2\epsilon \text{cost}_d(\mathcal{F}(X, d, k)) \quad (6b)$$

$$+ \sum_{i=1}^k \sum_{\{x,y\} \subseteq C_i} \frac{1}{|C_i|} 2\epsilon \quad (6c)$$

$$+ \sum_{i=1}^k \sum_{\{x,y\} \subseteq C_i} \frac{1}{|C_i|} \epsilon^2 \quad (6d)$$

While we cannot know the size of individual clusters in the general case we do know $\frac{1}{|C_i|}$ is upper bounded by 1. Therefore we can substitute $\frac{1}{|C_i|}$ with 1 in terms 6c and 6d while only increasing the value. For the same reasons we do not know the number of in-cluster point pairs in the general case in terms 6c and 6d. However we do know the number of in-cluster point pairs is bounded by the total number of point pairs, namely $\binom{n}{2}$ which can be substituted in the same way while only increasing the value.

$$\text{cost}_{d'}(\mathcal{F}(X, d', k)) \leq \text{cost}_d(\mathcal{F}(X, d, k)) \quad (7a)$$

$$+ 2\epsilon \text{cost}_d(\mathcal{F}(X, d, k)) \quad (7b)$$

$$+ \binom{n}{2} 2\epsilon \quad (7c)$$

$$+ \binom{n}{2} \epsilon^2 \quad (7d)$$

Therefore we know:

$$\text{cost}_{d'}(\mathcal{F}(X, d', k)) \leq (1 + 2\epsilon) \text{cost}_d(\mathcal{F}(X, d, k)) + \binom{n}{2} (2\epsilon + \epsilon^2)$$

Then, $c \geq 1 + 2\epsilon$, so $\epsilon \leq \frac{c-1}{2}$. Similarly, $c_0 \geq M(\epsilon^2 + 2\epsilon)$, so $\epsilon \leq \frac{-M + \sqrt{M^2 + 4MC_0}}{2M}$ where $M = \binom{n}{2}$. So, $\epsilon < \min(\frac{c-1}{2}, \frac{-M + \sqrt{M^2 + 4MC_0}}{2M})$. \square

Theorem 3. Consider the k -means clustering function and a data set (X, d) . If (X, d) is (δ, c, c_0, k) -uniquely optimal, then it is also (α, δ, k) -multiplicative perturbation robust for all $\alpha < \sqrt{c}$.

Proof. Consider a data set (X, d) , and let d' be any α -multiplicative perturbation of d .

First, we argue that $\text{cost}_{d'}(\mathcal{F}(X, d, k))$ is close to $\text{cost}_{d'}(\mathcal{F}(X, d', k))$. Let $C = \mathcal{F}(X, d, k)$. First, note that $\text{cost}_{d'}(\mathcal{F}(X, d', k)) \leq \text{cost}_{d'}(C)$. This is because \mathcal{F} finds the optimal solution on (X, d') , and so the clustering it selects can only have lower cost than the cost of C when evaluated with d' .

So, we calculate the cost of C on (X, d') . The k -means cost function is bounded by $\sum_{i=1}^k \sum_{x,y \in C_i} \frac{1}{|C_i|} d(x, y)^2$. After a multiplicative perturbation, the contribution of an edge of length $d(x, y)$, which used to contribute at most $d(x, y)^2$ to the cost of the function, contributes at most $(\alpha \cdot d(x, y))^2$, and so the contribution increases by at most a factor of α^2 . $\text{cost}_{d'}(\mathcal{F}(X, d')) \leq \text{cost}_{d'}(C) \leq \sum_i \sum_{\{x,y\} \subseteq C_i} \frac{1}{|C_i|} (\alpha \cdot d(x, y))^2 \leq \alpha^2 \text{cost}_d(C)$.

Then, $c \geq \alpha^2$, so $\alpha \leq \sqrt{c}$. \square

The proofs for k -medoid and min-sum follow similarly and are included in the appendix.

4.2 Perturbation robustness of Linkage-Based algorithms

We now move onto Linkage-Based algorithms, which in contrast to the methods studied in the previous section, do not seek to optimize an explicit objective function. Instead, they perform a series of merges, combining clusters according to their own measure of between-cluster distance.

Given clusters $A, B \subseteq X$, the following are the between-cluster distances of some of the most popular Linkage-Based algorithms:

- **Single linkage:** $\min_{a \in A, b \in B} d(a, b)$
- **Average linkage:** $\sum_{a \in A, b \in B} \frac{d(a, b)}{(|A| \cdot |B|)}$
- **Complete linkage:** $\max_{a \in A, b \in B} d(a, b)$

We consider Linkage-Based algorithms with the k -stopping criterion, which terminate an algorithm when k clusters remain, and return the resulting partitioning.

Because no explicit objective functions are used, we cannot rely on the uniqueness of optimum notion of clusterability. To define the type of cluster structure on which Linkage-Based algorithms exhibit perturbation robustness, we introduce a natural measure of clusterability based on a definition by Balcan et al Balcan et al. [2008b]. The original notion required data to contain a clustering where every element is closer to all elements in its cluster than to all other points. This notion was also used in Ackerman et al. [2012], Reyzin [2012], and Ackerman and Dasgupta [2014]. See Figure 4 for an illustration.

Definition 9 ((α, k) -strictly separable). A data set (X, d) is (α, k) -strictly separable if there exists a unique clustering $C = \{C_1, \dots, C_k\}$ of X so that for all $i \neq j$ and all $x, y \in C_i, z \in C_j$, $\alpha d(x, y) \leq d(x, z)$.

The definition for (ϵ, k) -strictly additive separable is analogous.

Definition 10 ((ϵ, k) -Strictly Additive Separable). A data set (X, d) is (ϵ, k) -Strictly Additive Separable if there exists a unique clustering $C = \{C_1, \dots, C_k\}$ of X so that for all $i \neq j$ and all $x, y \in C_i, z \in C_j$, $d(x, y) + \epsilon \leq d(x, z)$.

Before moving on to our results for Linkage-Based algorithms, we show that the above notions of clusterability are not sufficient to show that data is perturbation robust for k -means and similar methods. This indicates that different algorithms require different cluster structures in order to exhibit perturbation robustness. We show this results for (α, k) -strictly separable data. The proof for (ϵ, k) -strictly additive separable data is in the supplementary material.

Theorem 4. Let \mathcal{F} be any one of k -means, k -medoids, or min-sum. Then for any $\alpha > 1$, $\delta < \frac{2(k-1)n}{k^2(n-1)}$, there exists an (α, k) -strictly separable data set on which \mathcal{F} is not (α, δ) -multiplicative perturbation robust.

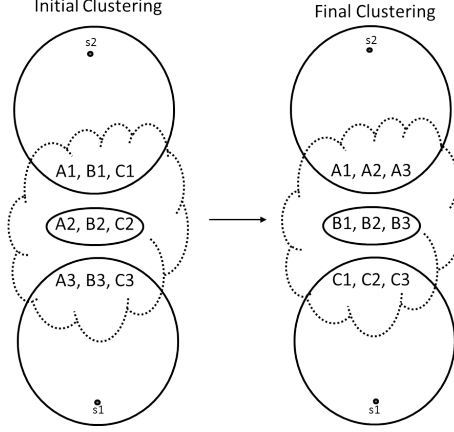


Figure 3: An illustration of the data set in the proof of Theorem 4 for $k = 3$. The original data consists of a dense region, “cloud” of points, and $k - 1$ outliers. Each of the A_i s, B_i s and C_i s consist of $\frac{n}{k^2}$ points. Before the perturbation, the data in the cloud is clustered $\{\{A_1 \cup B_1 \cup C_1\}, \{A_2 \cup B_2 \cup C_2\}, \{A_3 \cup B_3 \cup C_3\}\}$, whereas after the perturbation it is partitioned as $\{\{A_1 \cup A_2 \cup A_3\}, \{B_1 \cup B_2 \cup B_3\}, \{C_1 \cup C_2 \cup C_3\}\}$, leading to a large Hamming distance.

Proof. We construct such a data set (X, d) such that there is one cloud of points densely packed together and $k - 1$ singleton points far away from all other points. Further, in this construction $n|k^2$ and $n \gg k$. The strictly separable clustering of X will consist of the $k - 1$ singleton points being separate clusters and the cloud being in one cluster.

Arrange the cloud of points such that the ratio of the largest to smallest in-cloud distance is less than α , the ratio of the largest to smallest cloud point to singleton point distance is less than α , and all points are separated such that \mathcal{F} splits the cloud evenly into k separate clusters and the singleton points go into separate clusters.

Because the ratio between the largest and smallest in-cloud distance is less than α , an α -multiplicative perturbation can radically change the structure of the points in the cloud. If a point is identified by its distances to all other data, then perturbing the distance function can cause points to switch with one another. This ability to change applies similarly to the distance between cloud and singleton points. Because points can be arbitrarily made to act like other points, we perturb the data set such that the maximum number of in/between-cluster relationships are changed (with the restriction of never switching points from being in the cloud to being a singleton point and vice versa, because a multiplicative perturbation cannot necessarily make this switch).

We maximize the possible Hamming distance under the previous assumptions by constructing the following two data sets: First, divide the points of the cloud evenly into k clusters. This is our first clustering. Then taking that clustering, re-cluster the points by grouping the points in each cluster into groups of n/k^2 . Finally, form the new clusters by selecting one group from each previous cluster to be in a new cluster. See figure 3 for an example.

To find the Hamming distance between these two clusterings we first find the number of pairwise relationships that were formerly between-cluster that are now in-cluster. First, remember that a group contains n/k^2 points and there will be $k \binom{k}{2}$ group pairs that were formerly between-cluster that are now in cluster. Next, each point in a group will contribute n/k^2 to the Hamming distance per group pair. This gives the amount contributed to the Hamming distance by points relationships that were formerly between-cluster that are now in-cluster as $k \frac{n^2}{k^4} \binom{k}{2}$.

We now find the number of pairwise dissimilarities that were formerly in-cluster that are now between-cluster. Similar to before, each group contains n/k^2 points and there will be $k \binom{k}{2}$ groups that were formerly

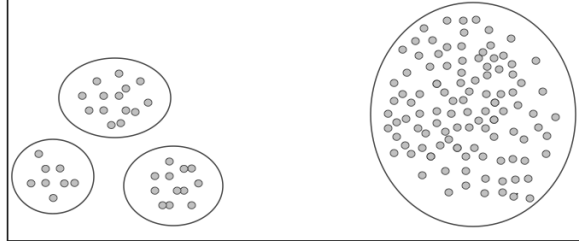


Figure 4: An example of strictly separable data. Note that it may include clusters with different diameters, as long as the dissimilarity between any two clusters scales as the larger diameter of the two.

in-cluster that are now between-cluster. This gives the total Hamming distance as $2k \frac{n^2}{k^4} \binom{k}{2} \binom{n}{2}^{-1}$, which reduces to $\frac{2(k-1)n}{k^2(n-1)}$. \square

We now show that whenever data is strictly separable, then it is also perturbation robust with respect to some of the most popular Linkage-Based algorithms. An analogous result for additive perturbation robustness appears in the supplementary material.

Theorem 5. *Single-Linkage, Average-Linkage, and Complete-Linkage are $(\alpha, 0)$ -multiplicative perturbation robust on all (α^2, k) -strictly separable data sets.*

Proof. We begin by showing that whenever data is $(1, k)$ -strictly separable, then these Linkage-Based algorithms identify the underlying cluster structure. This result was previously shown for Single-Linkage Balcan et al. [2008b] and Average-Linkage Ackerman et al. [2012]. We now prove this for complete-linkage.

First, we introduce the concept of a *refinement*. A clustering C' is a *refinement* of clustering C^* if C^* can be obtained by merging clusters in C' . The proof proceeds by induction on the number of iterations, showing that at each step of the algorithm, the current clustering is a refinement of the strictly separable k -clustering C . Since Linkage-Based algorithms start by placing each point in its own cluster, the clustering formed in the first step is a refinement of C . Assuming that the hypothesis holds at step i of the algorithm, we show that it is retained in the following step. Consider any C_1 and C_2 that are a subset of the same cluster in C , and any C_3 that is a subset of a different cluster in C . Let $(x, y) = \operatorname{argmax}_{x \in C_1, y \in C_2} d(x, y)$. Then, the dissimilarity between x and any point in C_3 is greater than $d(x, y)$ since the data is $(1, k)$ -strictly separable. Then Complete-Linkage merges C_1 with C_2 before merging C_1 with C_3 .

Lastly, observe that data that is (α^2, k) -strictly separable is also $(1, k)$ -strictly separable, and remains so after an α -perturbation, as shown in Lemma 1 in the appendix. It follows that single, average, and complete linkage are $(\alpha, 0)$ -Multiplicative Perturbation Robust on (α^2, k) -strictly separable data. \square

5 Conclusions

As a property of an algorithm, perturbation robustness fails in a strong sense, contradicting even more fundamental requirements of clustering functions. As such, no algorithm can exhibit this desirable characteristic on all data sets. Notably, this result persists even if we allow two-thirds of all pairwise distance to change following a perturbation.

However, a more optimistic picture emerges when considering clusterable data, and we show that popular paradigms are able to discover some cluster structures even on faulty data. Further, different clustering techniques are perturbation robust on different cluster structures. This has important implications for the “user’s dilemma,” which is the problem of selecting a suitable clustering algorithm for a given task. Faced with the challenge of clustering data with imprecise dissimilarities between pairwise entities, a user cannot

simply elect to apply a perturbation robust technique as no such methods exist, and as such the selection of suitable methods calls for some insight on the underlying structure of the data.

Future work will investigate robustness of heuristics, such as Lloyd's method, for which preliminary analysis suggests that the cluster structure required for perturbation robustness depends on the method of initialization.

Appendix

6 Impossibility theorems for $\delta = 0$

In prior work, perturbation robustness is often defined with $\delta = 0$, requiring that the clustering remain unchanged after a perturbation. Unfortunately, as a property of an algorithm, this formulation fails in a strong sense. That is, we show that no reasonable clustering function can satisfy this condition without effectively ignoring all pairwise distances, by outputting the same partitioning irrespective of the setting of d .

Specifically, we prove that both additive and multiplicative perturbation robustness with $\delta = 0$ contradicts 2-Richness, which is a relaxation Kleinberg's Richness axiom. This property requires clustering functions to be at least minimally responsive to pairwise dissimilarity. That is, given complete freedom to reassign all dissimilarities, we should be able to change the output of the function. This basic property is satisfied by all reasonable clustering methods Ackerman et al. [2010b].

Let $\text{Range}(\mathcal{F}, X, k)$ denote the set of all clusterings C so that $\mathcal{F}(X, d, k) = C$ for some dissimilarity function d .

Definition 11 (2-Richness). *For all X , $|\text{Range}(\mathcal{F}, X, k)| \geq 2$.*

We now prove that both additive and multiplicative perturbation robustness (with $\delta = 0$) are inconsistent with 2-Richness.

Theorem 6. *No clustering function is both 2-Rich and ϵ -Additive Perturbation Robust for any $\epsilon > 0$.*

Proof. Let \mathcal{F} be any 2-Rich clustering function. Then for any domain set X , there exist dissimilarity functions d and d' so that $\mathcal{F}(X, d, k) \neq \mathcal{F}(X, d', k)$. Observe that we can transform d into d' by making incremental changes, each changing the dissimilarity function by no more than ϵ on each pairwise dissimilarity. Then, by ϵ -Perturbation-Robustness, $\mathcal{F}(X, d, k) = \mathcal{F}(X, d', k)$, contradicting the previous claim. \square

Now we prove the analogous result for multiplicative perturbation robustness.

Theorem 7. *No clustering function is both 2-Rich and α -Multiplicative Perturbation Robust for any $\alpha > 1$.*

Proof. Let \mathcal{F} be any 2-Rich, Isomorphic Invariant, and α -Multiplicative Perturbation Robust clustering function. Then for any domain set X , there exist dissimilarity functions d and d' so that $\mathcal{F}(X, d, k) \neq \mathcal{F}(X, d', k)$. Observe we can transform d into d' through a series of α -multiplicative perturbations, changing each pairwise distance by a factor of α or less with each perturbation.

A contradiction is therefore achieved. By α -Multiplicative Perturbation Robustness $\mathcal{F}(X, d, k) = \mathcal{F}(X, d', k)$, and by 2-Richness $\mathcal{F}(X, d, k) \neq \mathcal{F}(X, d', k)$. \square

The above results implies that perturbation robustness with $\delta = 0$ is too stringent a requirement for clustering functions.

7 Proof of Lemma 1

Lemma 1. *The minimum separation a data set that is (α^2, k) -strictly separable can be after an α -multiplicative perturbation is $(1, k)$ -strictly separable.*

Proof. If a data set is (α^2, k) -strictly separable, the minimum between-cluster dissimilarity must be greater than the maximum in-cluster dissimilarity by a factor of α^2 , $\text{argmin}(d(x, y) | x, y \in Xx \not\sim_C y) \geq \alpha^2 \text{argmax}(d(x, y) | x, y \in Xx \sim_C y)$.

An α -multiplicative perturbation can change any pairwise distance by at most a factor of $\alpha^{\pm 1}$, $\text{argmin}(\alpha \cdot d(x, y) | x, y \in Xx \not\sim_C y) \geq \alpha^2 \text{argmax}(\alpha^{-1} \cdot d(x, y) | x, y \in Xx \sim_C y)$ because the multiplication is applied to

all arguments and is outside of the distance function this simplifies to $\alpha \cdot \operatorname{argmin}(d(x, y) | x, y \in X, x \not\sim_C y) \geq \alpha \cdot \operatorname{argmax}(d(x, y) | x, y \in X, x \sim_C y)$.

Therefore $\operatorname{argmin}(d(x, y) | x, y \in X, x \not\sim_C y) \geq \operatorname{argmax}(d(x, y) | x, y \in X, x \sim_C y)$. This is equivalent to the definition of $(1, k)$ -strictly separable. \square

8 Excluded multiplicative perturbation proofs

Theorem 8. *Consider the k -medoids clustering function and a data set (X, d) . If (X, d) is (δ, c, c_0, k) -uniquely optimal, then it is also (α, δ, k) -multiplicative perturbation robust for all $\alpha < c$.*

Proof. Consider a data set (X, d) , and let d' be any α -multiplicative perturbation of d .

First, we argue that $\operatorname{cost}_{d'}(\mathcal{F}(X, d, k))$ is close to $\operatorname{cost}_{d'}(\mathcal{F}(X, d', k))$. Let $C = \mathcal{F}(X, d, k)$. First, note that $\operatorname{cost}_{d'}(\mathcal{F}(X, d', k)) \leq \operatorname{cost}_{d'}(C)$. This is because \mathcal{F} finds the optimal solution on (X, d') , and so the clustering it selects can only have lower cost than the cost of C . We also compute the cost of each element to the same cluster centers as in C , as it provides an upper bound on $\operatorname{cost}_{d'}(C)$. So, $\operatorname{cost}_{d'}(C) \leq \alpha \cdot \operatorname{cost}_d(C)$. So, $c \geq \alpha$, so $\alpha < c$. \square

Theorem 9. *Given the min-sum objective, a data set that is $(\delta, \alpha, 0, k)$ -uniquely optimal is also (α, δ) -multiplicative perturbation robust.*

Proof. Let C be the optimal min-sum clustering of an arbitrary data set.

Let d' be any α -multiplicative perturbation of d . Therefore $\operatorname{cost}_{d'}(C)$ is at worst:

$$\begin{aligned} & \sum_{i=1}^k \sum_{x, y \in C_i} d'(x, y) \\ &= \sum_{i=1}^k \sum_{x, y \in C_i} \alpha d(x, y) \\ &= \alpha \cdot \operatorname{cost}_d(C) \end{aligned}$$

Because $\operatorname{cost}_{d'}(C) \leq \alpha \cdot \operatorname{cost}_d(C)$, $\Delta(\mathcal{F}(X, d, k), \mathcal{F}(X, d', k)) < \delta$. Therefore the maximum change an α -multiplicative perturbation can produce approaches δ . \square

9 Equivalent results for additive perturbation

We now prove the results in the main paper but for additive instead of multiplicative perturbation.

First, recall the definition of additive perturbation.

Definition 12 (ϵ -additive perturbation of a dissimilarity function). *Given a pair of dissimilarity functions d and d' over a domain X , d' is an ϵ -additive perturbation of d , for $\epsilon > 0$, if for all $x, y \in X$, $d(x, y) - \epsilon \leq d'(x, y) \leq d(x, y) + \epsilon$.*

Definition 13 $((\epsilon, \delta)$ -Additive Perturbation Robust Clustering Function). *A clustering function \mathcal{F} is (ϵ, δ) -Additive-Perturbation Robust, if for any d' that is an ϵ -additive-perturbation of d , $\Delta(\mathcal{F}(X, d, k), \mathcal{F}(X, d', k)) \leq \delta$.*

Theorem 10. *Consider the k -medoids clustering function. If (X, d) is (δ, c, c_0, k) -uniquely optimal, then it is also (ϵ, δ, k) -additive perturbation robust for all $\epsilon < \sqrt{\frac{2c_0}{n(n-1)}}$.*

Proof. Consider a data set (X, d) , and let d' be any ϵ -additive perturbation of d .

First, we argue that $\operatorname{cost}_{d'}(\mathcal{F}(X, d, k))$ is close to $\operatorname{cost}_{d'}(\mathcal{F}(X, d', k))$. Let $C = \mathcal{F}(X, d, k)$. First, note that $\operatorname{cost}_{d'}(\mathcal{F}(X, d', k)) \leq \operatorname{cost}_{d'}(C)$. This is because \mathcal{F} finds the optimal solution on (X, d') , and so the clustering it selects can only have lower cost than the cost of C . We also compute the cost of each element to the same cluster centers as in C , as it provides an upper bound on $\operatorname{cost}_{d'}(C)$. So, $\operatorname{cost}_{d'}(C) \leq \operatorname{cost}_d(C) + \epsilon \binom{n}{2}$. So, $c_0 \geq \epsilon \binom{n}{2}$, so $\epsilon < \sqrt{\frac{2c_0}{n(n-1)}}$. \square

Theorem 11. *Given the min-sum objective, $(\delta, 0, \epsilon \binom{n}{2}, k)$ -uniquely optimal data sets are (ϵ, δ) -additive perturbation robust.*

Proof. Let C be the optimal k -means clustering of an arbitrary data set.

Let d' be any ϵ -perturbation of d . Therefore $\text{cost}_{d'}(C)$ is at most:

$$\begin{aligned} & \sum_{i=1}^k \sum_{x,y \in C_i} d'(x,y) \\ & \leq \sum_{i=1}^k \sum_{x,y \in C_i} d(x,y) + \epsilon \\ & \leq \text{cost}_d(C) + \sum_{i=1}^k \sum_{x,y \in C_i} \epsilon \\ & \leq \text{cost}_d(C) + \epsilon \binom{n}{2} \end{aligned}$$

Because $\text{cost}_{d'}(C) < \text{cost}_d(C) + \epsilon \binom{n}{2}$, $\Delta(\mathcal{F}(X, d, k), \mathcal{F}(X, d', k)) < \delta$. Therefore the maximum change an ϵ -perturbation can produce approaches δ . \square

Definition 14 ((ϵ, k) -Strictly Additive Separable). *A data set (X, d) is (ϵ, k) -Strictly Additive Separable if there exists a unique clustering $C = \{C_1, \dots, C_k\}$ of X so that for all $i \neq j$ and all $x, y \in C_i$, $z \in C_j$, $d(x, y) + \epsilon \leq d(x, z)$.*

Theorem 12. *Let \mathcal{F} be any one of k -means, k -medoids, or min-sum. Then for any $\epsilon > 0$, $\delta < \frac{(k-2)(n-k)^2}{(k-1)n^2}$, there exists an (ϵ, k) -strictly-additive separable data set on which \mathcal{F} is not (ϵ, δ) -additive perturbation robust.*

Proof. With respect to ϵ -additive perturbations:

Taking the same base data set as before we now set the cloud to be sufficiently dense, and the singleton points to be organized so that \mathcal{F} merges all but the large cluster, and as such the final clustering depends nearly entirely on the internal structure of the large cluster.

Arrange the the data set to consist of dissimilarities smaller than ϵ , so an ϵ -perturbation of d can radically alter the output of \mathcal{F} . In particular, we can arrange the large cluster so that $\mathcal{F}(X, d, k)$ subdivides it into k equal size groups, and after the perturbation all points in that cluster form a single cluster by moving the points in that cluster very close together.

We now compute the distance between these two clusterings. The original clustering, looking only at the data in the large cluster, has $\binom{k-1}{2} \left(\frac{n-k+1}{2}\right)^2$ between-cluster edges, all of which become incluster edges after the perturbation. So, the two clusterings differ by at least $\binom{k-1}{2} \left(\frac{n-k+1}{k-1}\right)^2 / \binom{n}{2} > \frac{(k-2)(n-k)^2}{(k-1)n^2}$. \square

Theorem 13. *Single-Linkage, Average-Linkage, and Complete-Linkage are $(\epsilon, 0)$ -perturbation robust on all $(2\epsilon, k)$ -strictly additive Separable data sets.*

Proof. We begin by showing that whenever data is $(0, k)$ -strictly additive separable, then these Linkage-Based algorithms identify the underlying cluster structure. This result was previously shown for Single-Linkage and Average-Linkage. We now prove this for complete-linkage.

Recall the definition of a *refinement*. A clustering C' is a *refinement* of clustering C^* if C^* can be obtained by merging clusters in C' . The proof proceeds by induction on the number of iterations, showing that at each step of the algorithm, the current clustering is a refinement of the strictly separable k -clustering C . Since Linkage-Based algorithms start by placing each point in its own cluster, the clustering formed in the first step is a refinement of C . Assuming that the hypothesis holds at step i of the algorithm, we show that it is retained in the following step. Consider any C_1 and C_2 that are a subset of the same cluster in C , and any C_3 that is a subset of a different cluster in C . Let $(x, y) = \arg\max_{x \in C_1, y \in C_2} d(x, y)$. Then, the

dissimilarity between x and any point in C_3 is greater than $d(x, y)$ since the data is $(0, k)$ -strictly separable. Then Complete-Linkage merges C_1 with C_2 before merging C_1 with C_3 .

Lastly, observe that $(2\epsilon, k)$ -strictly additive separable data is also $(0, k)$ -strictly additive separable, and remains so after an ϵ -perturbation. It follows that Single-Linkage, Average-Linkage, and Complete-Linkage are $(\epsilon, 0)$ -Perturbation-Robust on $(2\epsilon, k)$ -strictly separable data. See lemma below. \square

Lemma 2. *The minimum separation a data set that is $(2\epsilon, k)$ -strictly additive separable can be after an ϵ -additive perturbation is $(0, k)$ -strictly additive separable*

Proof. If a data set is $(2\epsilon, k)$ -strictly additive separable then the maximum in cluster distance is smaller than the minimum out of cluster distance by 2ϵ , $\argmin(d(x, y)|x, y \in X \text{ } x \not\sim_c y) - \argmax(d(x, y)|x, y \in X \text{ } x \sim_c y) = 2\epsilon$. An ϵ -additive perturbation can change any pairwise distance by at most $\pm\epsilon$, $\argmin(d(x, y) + \epsilon|x, y \in X \text{ } x \not\sim_c y) - \argmax(d(x, y) - \epsilon|x, y \in X \text{ } x \sim_c y) = 2\epsilon$. because addition and subtraction is applied to all arguments this simplifies to $\argmin(d(x, y)|x, y \in X \text{ } x \not\sim_c y) + \epsilon - (\argmax(d(x, y)|x, y \in X \text{ } x \sim_c y) - \epsilon) = 2\epsilon$. Therefore, $\argmin(\argmin(d(x, y)|x, y \in X \text{ } x \not\sim_c y) + \epsilon - \argmax(d(x, y) \pm \frac{\epsilon}{r}|x, y \in X \text{ } x \sim_c y)) = 0$. This is equivalent to being $(0, k)$ -strictly additive separable. \square

10 Impossibility Result for Euclidean Space Clustering

Definition 15 (Euclidean α -multiplicative perturbation). *Given a data set embedded into Euclidean space X , an α -multiplicative perturbation of X produces X' s.t. all $x \in X$ can be moved freely in the space to produce x' given that for all $x, y \in X$, $x', y' \in X'$, $\alpha^{-1} \cdot d(x, y) \leq d(x', y') \leq \alpha \cdot d(x, y)$.*

Definition 16 (Euclidean ϵ -additive perturbation). *Give a data set embedded into Euclidean space X , an ϵ -additive perturbation of X produces X' s.t. all $x \in X$ can be moved freely in the space to produce x' given that for all $x, y \in X$, $x', y' \in X'$, $d(x, y) - \epsilon \leq d(x', y') \leq d(x, y) + \epsilon$.*

Theorem 14. *There can be no clustering algorithm that follows the three body rule, is data replication invariant and is either α -multiplicative perturbation robust or ϵ -additive perturbation robust while clustering data in Euclidean space.*

Proof. Like before we create a data set $X = \{a, b, c\}$. However this time we define where the points lie in Euclidean space. Let $a = \vec{1}_d$ Where $\vec{1}_d$ is the d -dimensional vector with all ones. Next we define $b = 2 + \epsilon'_d$ for some arbitrarily small ϵ' and $c = \overrightarrow{3 + \epsilon'_d}$. By the three body rule the resulting clustering must be $C = \{\{a\}, \{b, c\}\}$. Now, we perturb the points to create X' . The α -multiplicative perturbation moves the point b to $\overrightarrow{\frac{2+\epsilon'}{\alpha}_d}$, and the ϵ -additive perturbation moves the point b to $\overrightarrow{(2 + \epsilon' - \epsilon)_d}$ both leaving all other points untouched. Neither of the perturbations change any pairwise distance by a factor of $\alpha^{\pm 1}$ or in absolute $\pm\epsilon$ respectively.

Note in the multiplicative case the ϵ' addition can be set arbitrarily small s.t. the α^{-1} factor will dominate whether the point is further or closer to the $\vec{0}_d$ than $\vec{2}_d$.

By the three body rule X' must be clustered as $C = \{\{a, b\}, \{c\}\}$ and the perturbation robustness demands the clustering be $C = \{\{a\}, \{b, c\}\}$. Finally, data replication invariance can be used to increase the size of X' to be arbitrarily large no longer requiring there to be only three points while still maintaining the contradiction. \square

References

- M. Ackerman and S. Ben-David. Clusterability: A theoretical study. *Proceedings of AISTATS-09, JMLR: W&CP*, 5(1-8):53, 2009.
- M. Ackerman and S. Dasgupta. Incremental clustering: The case for extra clusters. In *Advances in Neural Information Processing Systems*, pages 307–315, 2014.
- M. Ackerman, S. Ben-David, and D. Loker. Characterization of linkage-based clustering. In *Proceedings of the Twenty-third Conference on Learning Theory (COLT 2010)*. COLT, 2010a.
- M. Ackerman, S. Ben-David, and D. Loker. Towards property-based classification of clustering paradigms. In *Advances in Neural Information Processing Systems*. NIPS, 2010b.
- M. Ackerman, S. Ben-David, S. Branzei, and D. Loker. Weighted clustering. *Proc. 26th AAAI Conference on Artificial Intelligence*, 2012.
- M. Ackerman, S. Ben-David, D. Loker, and S. Sabato. Clustering oligarchies. *Proceedings of AISTATS-12, JMLR: W&CP*, 31(6674), 2013.
- M. Agarwal, R. Jaiswal, and A. Pal. k-means++ under approximation stability. In *Theory and Applications of Models of Computation*, pages 84–95. Springer, 2013.
- P. Awasthi, A. Blum, and O. Sheffet. Center-based clustering under perturbation stability. *Information Processing Letters*, 112(1):49–54, 2012.
- M. Balcan, A. Blum, and S. Vempala. A discriminative framework for clustering via similarity functions. In *Proceedings of the 40th annual ACM symposium on Theory of Computing*, pages 671–680. ACM, 2008a.
- M.-F. Balcan, A. Blum, and S. Vempala. A discriminative framework for clustering via similarity functions. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 671–680. ACM, 2008b.
- S. Ben-David. Computational feasibility of clustering under clusterability assumptions. *arXiv preprint arXiv:1501.00437*, 2015.
- S. Ben-David and L. Reyzin. Data stability in clustering: A closer look. *Theoretical Computer Science*, 558: 51–61, 2014.
- Y. Bilu and N. Linial. Are stable instances easy. In *1st Symposium on Innovations in Computer Science (ICS)*, 2010.
- L. Fisher and J. Van Ness. Admissible clustering procedures. *Biometrika*, 58(1):91–104, 1971.
- N. Jardine and R. Sibson. Mathematical taxonomy. *London*, 1971.
- J. Kleinberg. An impossibility theorem for clustering. In *Advances in Neural Information Processing Systems*. NIPS, 2003.
- R. Ostrovsky, Y. Rabani, L. Schulman, and C. Swamy. The effectiveness of Lloyd-type methods for the k -means problem. In *Foundations of Computer Science, 2006. FOCS’06. 47th Annual IEEE Symposium on*, pages 165–176, 2006.
- L. Reyzin. Data stability in clustering: A closer look. In *Algorithmic Learning Theory*, pages 184–198. Springer, 2012.
- S. Sahni and T. Gonzalez. P-complete approximation problems. *Journal of the ACM (JACM)*, 23(3):555–565, 1976.

- D. Steinley. K-means clustering: a half-century synthesis. *British Journal of Mathematical and Statistical Psychology*, 59(1):1–34, 2006.
- W. Wright. A formalization of cluster analysis. *Pattern Recognition*, 5(3):273–282, 1973.
- R. Zadeh and S. Ben-David. A uniqueness theorem for clustering. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. UAI, 2009.